Analyzing Testing ROI

Quantifiable Ways Testing Saves Money

Champion bull, Deets, photo © by and used with permission of of The Double D Ranch, thedoubled@aol.com

Black bear photo © by and used with permission of of Wild Things Photography, tah@wildthingsphoto.com

RBCS

TIME TESTED.
TESTING IMPROVED.

www.RBCS-US.com

Testing Doesn't Cost, It Saves

- A well-established concept
  - In *Quality Is Free*, Crosby argued that economic benefits of quality always exceed costs
  - In *Quality Control Handbook*, Juran and Gryna explained the cost of quality model
- Does this hold for software?
  - Slaughter, Harter, and Krishnan in "Evaluating the Cost of Software Quality"
  - Campanella, et al., in *Principles of Quality Costs*, especially the Raytheon Electronic Systems group case study
- If testing returns more to the company than it costs, can we quantify these benefits?

Four Ways Testing Saves Money

- Smart test groups, employing effective and efficient test techniques, can save money in four ways:
  - Finding bugs that get fixed—or even preventing them
  - Finding bugs that don't get fixed—but are known
  - Running tests that mitigate (potentially expensive) risks
  - Guiding the project to success through timely, accurate, credible information for project tracking
- Let's see how…

Finding Bugs that Get Fixed

- The idea that it's cheaper to find and fix bugs before release is formalized in a concept called cost of (poor) quality
  - $C_{quality} = C_{conformance} + C_{nonconformance}$
  - Conformance costs include testing (finding bugs, a.k.a. defects) and quality assurance (preventing bugs)
  - Nonconformance costs include fixing bugs, retesting, dealing with angry customers, damage to company image, lost business, etc.
- Can testing reduce the cost of quality?
  - Let's look at a hypothetical case study
  - Conformance costs: manual, automated, static testing only (no QA)
  - Nonconformance costs: fixing bugs only (no intangibles)
  - Assume 1,000 must-fix bugs in each (quarterly) release
  - Assume following bug-fix costs: $1 in review, $10 in programmer testing, $100 in tester testing, and $1,000 in customer usage

No Formal Testing

- 1,000 bugs in product
- Developers find 250
- Customers find 750
- Value added by testing:
  - None yet—but the opportunity is clear
  - Quality costs over 3/4 million dollars…
  - …and customers are angry!
- Let's introduce formal manual testing with an independent test team…

Testing Investment Options: ROI Analysis				
	No Formal			
Testing	**Testing**			
Staff	$0			
Infrastructure	0			
Tools	0			
Total Investment	**$0**			
Development				
Must-Fix Bugs Found	250			
Fix Cost	2,500			
Testing				
Must-Fix Bugs Found	0			
Fix Cost	0			
Customer Support				
Must-Fix Bugs Found	750			
Fix Cost	750,000			
Cost of Quality				
Conformance	**$0**			
Nonconformance	**$752,500**			
Total CoQ	**$752,500**			

Manual Testing

- Invest in testing team
- Developers find 250 bugs
- Testers find 350 bugs
- Value added:
  - Quality costs down ~1/3
  - Test return on investment (ROI): 350%
  - Customers find ~40% fewer bugs
- Suppose we can find about 40% more bugs with test automation…

Testing Investment Options: ROI Analysis

	No Formal Testing	Manual Testing
Testing		
Staff	$0	$60,000
Infrastructure	0	10,000
Tools	0	0
Total Investment	**0**	**70,000**
Development		
Must-Fix Bugs Found	250	250
Fix Cost	2,500	2,500
Testing		
Must-Fix Bugs Found	0	350
Fix Cost	0	35,000
Customer Support		
Must-Fix Bugs Found	750	400
Fix Cost	750,000	400,000
Cost of Quality		
Conformance	**$0**	**$70,000**
Nonconformance	**$752,500**	**$437,500**
Total CoQ	**$752,500**	**$507,500**
Return on Investment	**#N/A**	**350%**

Manual and Automated Testing

- Invest $150,000 in tools
  - Amortized over twelve quarterly releases
- Complement manual testing with automation
- Value added by testing:
  - Quality costs halved
  - Testing ROI: 445%
  - Customers find ~66% fewer bugs
- Maybe we can prevent bugs, not just detect them...

Testing Investment Options: ROI Analysis

Testing	No Formal Testing	Manual Testing	Automated Testing
Staff	$0	$60,000	$60,000
Infrastructure	0	10,000	10,000
Tools (amortized)	0	0	12,500
Total Investment	0	70,000	82,500
Development			
Must-Fix Bugs Found	250	250	250
Fix Cost	2,500	2,500	2,500
Testing			
Must-Fix Bugs Found	0	350	500
Fix Cost	0	35,000	50,000
Customer Support			
Must-Fix Bugs Found	750	400	250
Fix Cost	750,000	400,000	250,000
Cost of Quality			
Conformance	**$0**	**$70,000**	**$82,500**
Nonconformance	**$752,500**	**$437,500**	**$302,500**
Total CoQ	**$752,500**	**$507,500**	**$385,000**
Return on Investment	**#N/A**	**350%**	**445%**

Manual, Automated, and Static Testing

- Testers review design and requirements specs
  - Testers ask smart questions
  - Prevent 150 bugs
- Value added:
  - Quality costs down by 2/3
  - Testing ROI: 627%
  - Customers find 90% fewer bugs
- ➢ Suppose testers find an additional 650 real—but not must-fix—bugs…

Testing Investment Options: ROI Analysis

Testing	No Formal Testing	Manual Testing	Automated Testing	Static Testing
Staff	$0	$60,000	$60,000	$60,000
Infrastructure	0	10,000	10,000	10,000
Tools (amortized)	0	0	12,500	12,500
Total Investment	0	70,000	82,500	82,500
Development				
Must-Fix Bugs Found	250	250	250	250
Fix Cost	2,500	2,500	2,500	2,500
Testing				
Must-Fix Bugs Found	0	350	500	650
Fix Cost	0	35,000	50,000	50,150
Customer Support				
Must-Fix Bugs Found	750	400	250	100
Fix Cost	750,000	400,000	250,000	100,000
Cost of Quality				
Conformance	**$0**	**$70,000**	**$82,500**	**$82,500**
Nonconformance	**$752,500**	**$437,500**	**$302,500**	**$152,650**
Total CoQ	**$752,500**	**$507,500**	**$385,000**	**$235,150**
Return on Investment	**#N/A**	**350%**	**445%**	**627%**

Finding Bugs that Don't *Get Fixed?*

- So, you're thinking: What the heck good is *that*!?!
- Well, if we know where the bugs are, we can…
  1. Prevent the user from seeing the bug
  2. Warn users in release notes so they can avoid the bug
  3. Provide workaround and other info to help/support
- What are items 1, 2, and 3 worth?
- Let's return to our hypothetical case study…

Important Tips for this Bit of the ROI Equation:

1. Testing should, on average, identify must-fix bugs before non-must-fix bugs.
2. A cross-functional bug triage process should place bugs into each category.

ROI for Bugs that Don't Get Fixed

- Suppose the help desk or technical support staff know that, on average (with significant variation around the mean):
  - A call for a known bug is 15 minutes shorter than a call for an unknown bug
  - Each bug generates five calls to tech support
  - A tech support person costs the organization $40 per hour
- Then, each bug found, even if not fixed, saves (based on item 3 alone) :

$$40 \times \frac{15}{60} \times 5 = \$50$$

Challenge: Quantify the benefit of finding the bugs users are preventing from seeing (item 1) or manage to avoid (item 2). Perhaps you can estimate the tech support calls that *didn't* happen? How about user time that *didn't* get wasted?

Bug Information ROI Analysis

- Testers are finding the non-must-fix bugs already
  - No increased investment
- Value added:
  - $32,500 worth of tech support time saved
  - Testing ROI: 666%
  - Tech support people are happier…
- And testing adds *more* value...

Testing Investment Options: ROI Analysis

Testing	No Formal Testing	Manual Testing	Automated Testing	Static Testing	Bug Info Generated
Staff	$0	$60,000	$60,000	$60,000	$60,000
Infrastructure	0	10,000	10,000	10,000	10,000
Tools (amortized)	0	0	12,500	12,500	12,500
Total Investment	0	70,000	82,500	82,500	82,500
Development					
Must-Fix Bugs Found	250	250	250	250	250
Fix Cost	2,500	2,500	2,500	2,500	2,500
Testing					
Must-Fix Bugs Found	0	350	500	650	650
Fix Cost	0	35,000	50,000	50,150	50,150
Non-Must-Fix Bugs					650
Customer Support					
Must-Fix Bugs Found	750	400	250	100	100
Fix Cost	750,000	400,000	250,000	100,000	100,000
Tech Time Savings					32,500
Cost of Quality					
Conformance	**$0**	**$70,000**	**$82,500**	**$82,500**	**$82,500**
Nonconformance	**$752,500**	**$437,500**	**$302,500**	**$152,650**	**$152,650**
Total CoQ	**$752,500**	**$507,500**	**$385,000**	**$235,150**	**$235,150**
Return on Investment	**#N/A**	**350%**	**445%**	**627%**	**666%**

Testing is like Insurance

- Testing reduces what risk managers call *the cost of exposure*
  - Testing manages risks to quality, which have financial consequences
  - We buy insurance to mitigate serious financial risks
- Consider the economic concept of substitution
  - Insurance: a statistical mechanism for pooling risk
  - Conceptually there's no reason why that pooled risk couldn't be the risks to the quality of a system being developed
- Can we quantify the value of a test risk mitigation?
  - Insurance model: Expected payout = probability of loss * cost of loss
  - Value of test: The cost of likely field failures in tested behaviors
- ➢ Going back to the hypothetical case study…

Substitution: use of an alternative product or service that provides similar utility

Insurance as a Substitute for Testing

- Suppose we estimate the following factors for quality risks based on similar (technical, organization, business) projects:
  - Performance problems: $100,000 average loss, 10% likelihood
  - Functionality problems: $5,000 average loss, 50% likelihood
  - Security problems: $250,000 average loss, 5% likelihood
  - Other problems: $10,000 average loss, 10% likelihood
- Then, insurance for these risks would cost at least:

$$100{,}000 \times .1 + 5{,}000 \times .5 + 250{,}000 \times .05 + 10{,}000 \times .1 = \$26{,}000$$

Three Caveats to this Method:

1) Average costs and likelihoods probably based on a small sample of projects.
2) Some organizations are less risk-averse than others, and wouldn't buy insurance for risks to system quality even at a fair—or bargain—price.
3) The testers won't pay the project team for external failure (i.e., missed bugs).

Insurance ROI Analysis

- Testers are testing these areas already
  - No increased investment
- Value added:
  - $26,000 worth of "insurance"
  - Testing ROI: 698%
  - Management is more comfortable…
- And testing creates information…

Testing Investment Options: ROI Analysis

Testing	No Formal Testing	Manual Testing	Automated Testing	Static Testing	Bug Info Generated	Exposure Reduced
Staff	$0	$60,000	$60,000	$60,000	$60,000	$60,000
Infrastructure	0	10,000	10,000	10,000	10,000	10,000
Tools (amortized)	0	0	12,500	12,500	12,500	12,500
Total Investment	0	70,000	82,500	82,500	82,500	82,500
Development						
Must-Fix Bugs Found	250	250	250	250	250	250
Fix Cost	2,500	2,500	2,500	2,500	2,500	2,500
Testing						
Must-Fix Bugs Found	0	350	500	650	650	650
Fix Cost	0	35,000	50,000	50,150	50,150	50,150
Non-Must-Fix Bugs					650	650
"Insurance" Provided						26,000
Customer Support						
Must-Fix Bugs Found	750	400	250	100	100	100
Fix Cost	750,000	400,000	250,000	100,000	100,000	100,000
Tech Time Savings					32,500	32,500
Cost of Quality						
Conformance	**$0**	**$70,000**	**$82,500**	**$82,500**	**$82,500**	**$82,500**
Nonconformance	**$752,500**	**$437,500**	**$302,500**	**$152,650**	**$152,650**	**$152,650**
Total CoQ	**$752,500**	**$507,500**	**$385,000**	**$235,150**	**$235,150**	**$235,150**
Return on Investment	**#N/A**	**350%**	**445%**	**627%**	**666%**	**698%**

Testing Information Guides Project Success

- In *Estimating Software Costs*, Capers Jones identifies poor project tracking as a key cause of project failure
  - Accurate, credible, timely testing metrics (defects, tests completed, etc.) are a key part of the project information needed for tracking
  - We can extrapolate Jones' figures for a medium-sized project: good project tracking reduces the risk of project failure from 40% to 20%
  - If good testing and test metrics can provide half of this risk reduction benefit, then we can claim 10% of the value to the project
- For a final look at the ROI in our hypothetical case study…

Caveat: You'll need to adjust these numbers based on the size of the project you're working on, because the figures vary significantly. Jones quotes a 2% risk of project failure for very small projects all the way up to an 85% risk of project failure for extremely large projects!

ROI for Test-Related Tracking Information

- Assume our hypothetical case study is a medium-sized project
  - The information generated by testing will contribute 10% to reducing the risk of project failure
  - The project cost is the only value at risk (i.e., we ignore the opportunity costs of project failure)
  - The project costs are the testing and development budgets ($82,500 and $247,500, respectively)
- So, the value of the tracking information is:

$$(82{,}500 + 247{,}500) \times 0.1 = \$33{,}000$$

RBCS
TIME TESTED. TESTING IMPROVED.

Bug Information ROI Analysis

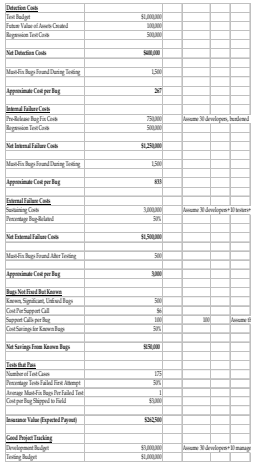
- Test results must be a credible, accurate, timely source of project information
- Value added:
  - A ten percent reduction in project failure risk (worth $33,000)
  - Testing ROI: 738%
  - Management is informed…
- A respectable business case…

Testing Investment Options: ROI Analysis

Testing	No Formal Testing	Manual Testing	Automated Testing	Static Testing	Bug Info Generated	Exposure Reduced	Project Tracking
Staff	$0	$60,000	$60,000	$60,000	$60,000	$60,000	$60,000
Infrastructure	0	10,000	10,000	10,000	10,000	10,000	10,000
Tools (amortized)	0	0	12,500	12,500	12,500	12,500	12,500
Total Investment	0	70,000	82,500	82,500	82,500	82,500	82,500
Development							
Must-Fix Bugs Found	250	250	250	250	250	250	250
Fix Cost	2,500	2,500	2,500	2,500	2,500	2,500	2,500
Development Investment							247,500
Testing							
Must-Fix Bugs Found	0	350	500	650	650	650	650
Fix Cost	0	35,000	50,000	50,150	50,150	50,150	50,150
Non-Must-Fix Bugs					650	650	650
"Insurance" Provided						26,000	26,000
Project Tracking Value							33,000
Customer Support							
Must-Fix Bugs Found	750	400	250	100	100	100	100
Fix Cost	750,000	400,000	250,000	100,000	100,000	100,000	100,000
Tech Time Savings					32,500	32,500	32,500
Cost of Quality							
Conformance	**$0**	**$70,000**	**$82,500**	**$82,500**	**$82,500**	**$82,500**	**$82,500**
Nonconformance	**$752,500**	**$437,500**	**$302,500**	**$152,650**	**$152,650**	**$152,650**	**$152,650**
Total CoQ	**$752,500**	**$507,500**	**$385,000**	**$235,150**	**$235,150**	**$235,150**	**$235,150**
Return on Investment	**#N/A**	**350%**	**445%**	**627%**	**666%**	**698%**	**738%**

Case Study Testing ROI

Click on the thumbnail to walk through a case study of testing ROI from an Internet appliance test project



Detection Costs					
Test Budget	$1,000,000				
Future Value of Assets Created	100,000				
Regression Test Costs	500,000				
Net Detection Costs	**$600,000**				
Must-Fix Bugs Found During Testing	1,500				
Approximate Cost per Bug	**267**				
Internal Failure Costs					
Pre-Release Bug Fix Costs	750,000		Assume 30 developers, burdened		
Regression Test Costs	500,000				
Net Internal Failure Costs	**$1,250,000**				
Must-Fix Bugs Found During Testing	1,500				
Approximate Cost per Bug	**833**				
External Failure Costs					
Sustaining Costs	3,000,000		Assume 30 developers+10 testers+		
Percentage Bug-Related	50%				
Net External Failure Costs	**$1,500,000**				
Must-Fix Bugs Found After Testing	500				
Approximate Cost per Bug	**3,000**				
Bugs Not Fixed But Known					
Known, Significant, Unfixed Bugs	500				
Cost Per Support Call	$0				
Support Calls per Bug	100		100	Assume 0	
Cost Savings for Known Bugs	50%				
Net Savings From Known Bugs	**$150,000**				
Tests that Pass					
Number of Test Cases	1725				
Percentage Tests Failed First Attempt	50%				
Average Must-Fix Bugs Per Failed Test	1				
Cost per Bug Shipped to Field	$3,000				
Insurance Value (Expected Payout)	**$262,500**				
Good Project Tracking					
Development Budget	$3,000,000		Assume 30 developers+10 manage		
Testing Budget	$1,000,000				

DO Try this at Home

- You can get a detailed walkthrough on this technique and the case study in the Basic Library at www.rbcs-us.com, in the article called "Testing ROI: What IT Managers Should Know"
- Go back to your office, give it a try, and see what you find
- I'll bet you're saving the company more money than you ever imagined

…Contact RBCS

For over a dozen years, RBCS has delivered services in consulting, outsourcing and training for software and hardware testing. Employing the industry's most experienced and recognized consultants, RBCS conducts product testing, builds and improves testing groups and hires testing staff for hundreds of clients worldwide. Ranging from Fortune 20 companies to start-ups, RBCS clients save time and money through improved product development, decreased tech support calls, improved corporate reputation and more. To learn more about RBCS, visit www.rbcs-us.com.

Address:	RBCS, Inc. 31520 Beck Road Bulverde, TX 78163-3911 USA
Phone:	+1 (830) 438-4830
Fax:	+1 (830) 438-4831
E-mail:	info@rbcs-us.com
Web:	www.rbcs-us.com